PEDAL INSIGHTS

RECOMENDACIONES INTELIGENTES, DECISIONES ACERTADAS.

Presentado a:
DS ARKON

Presentado por:
JOSE FERNANDO GARCIA VIDAL

18 Diciembre 2024

TABLA DE CONTENIDO

1. ANÁLISIS EXPLORATORIO Y PREPARACIÓN DE DATOS

El presente proyecto aborda el análisis de un sistema de bicicletas compartidas, con el objetivo de comprender los patrones de uso, características de los usuarios y optimizar la base de datos para su aplicación en modelos predictivos. Se trabajó con dos datasets principales: **entrenamiento** y **prueba**, los cuales fueron sometidos a una exploración detallada, limpieza rigurosa e ingeniería de características.

El conjunto de datos inicial comprende *1,269,886 registros* distribuidos en:

- Conjunto de Entrenamiento: 700,000 registros.
- Conjunto de Prueba: 569,886 registros.

Estructura del Dataset Original: Las variables disponibles incluyen:

- Variables Temporales: start_time, end_time
- Variables Geoespaciales: start_lat, start_lon, end_lat, end_lon
- Variables Identificadoras: trip_id, bike_id, start_station, end_station
- Variables de Duración: duration, plan_duration
- Categóricas: trip_route_category, passholder_type

Detección y Tratamiento de Valores Nulos

Columna	Valores Nulos	Porcentaje
plan_duration	208	0.03%
bike_id	17	0.002%

Decisión sobre Valores Nulos

Se evaluaron dos enfoques:

1. **Eliminación de registros incompletos**: Al representar menos del 3% del total, su impacto en la base de datos era mínimo.
2. **Imputación**: Aunque la imputación por media o mediana fue considerada, introducir supuestos adicionales podría sesgar el análisis.

Decisión Final: Se optó por eliminar los registros nulos para garantizar la coherencia y simplicidad del pipeline de limpieza, resultando en un dataset **depurado** con **675,618 registros**.

2. INGENIERÍA DE CARACTERÍSTICAS (FEATURE ENGINEERING)

La **ingeniería de características** desempeñó un papel fundamental en la transformación del dataset inicial, permitiendo capturar información crítica a partir de las variables originales. Este proceso incluyó la creación de métricas **temporales**, **geoespaciales** y **de desempeño agregado**, las cuales proporcionan una visión más detallada y enriquecida de los patrones de uso del sistema.

Las nuevas características generadas no solo facilitan la interpretación de los datos, sino que también aseguran un modelo más robusto y preciso al capturar tendencias temporales, eficiencia de los viajes y preferencias de los usuarios.

2.1 Variables Temporales

Se generaron variables que permiten identificar patrones de comportamiento a lo largo del tiempo, tales como franjas horarias, días de la semana y estacionalidad.

Variable	Descripción	Objetivo
franja_horaria	Categorización de start_time en franjas del día: Mañana, Tarde, Noche, Madrugada.	Identificar los picos de demanda en diferentes momentos del día.
estacion_año	Clasificación del mes de inicio del viaje en primavera, verano, Otoño e Invierno.	Capturar la **estacionalidad** en los patrones de uso.
dia_semana	Día de la semana derivado de start_time (0 = lunes, 6 = Domingo).	Analizar la **frecuencia de viajes** por día de la semana.
hora_inicio	Hora del día extraída de start_time.	Identificar patrones de uso en horas específicas.

Estas características permiten analizar tendencias temporales, tales como la alta demanda en horarios pico laborales y la estacionalidad durante el verano y otoño.

2.2 Variables Geoespaciales

Se calcularon métricas espaciales a partir de las coordenadas de inicio y fin de los viajes (start_lat, start_lon, end_lat, end_lon).

Variable	Descripción	Objetivo
distancia_mts	Distancia en **metros** entre las estaciones de inicio y fin del viaje, calculada mediante geometría de GeoPandas.	Medir la eficiencia de los viajes y diferenciar entre trayectos cortos y largos.

La variable distancia_mts es clave para entender la eficiencia del sistema, así como para identificar posibles patrones en trayectos más frecuentes entre estaciones cercanas.

2.3 Variables de Desempeño y Agregadas

Se agregaron variables derivadas de la duración, eficiencia y patrones de uso del sistema para proporcionar mayor profundidad en el análisis de desempeño.

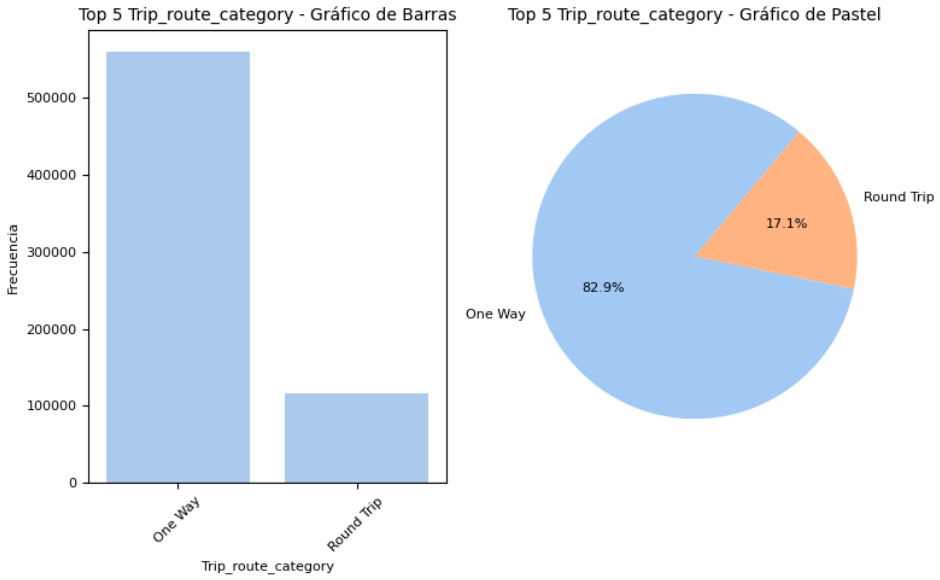
Variable	**Descripción**	**Objetivo**
tiempo_promedio_estacion	Duración promedio de los viajes iniciados en cada estación.	Identificar estaciones con mayor o menor eficiencia.
tiempo_promedio_pares	Duración promedio entre pares de estaciones (inicio y fin).	Capturar patrones de viajes frecuentes.
velocidad_promedio_mts	Relación entre distancia_mts y la duración del viaje (m/min).	Evaluar la **eficiencia** del viaje.
duration_normalized	Normalización de la duración del viaje respecto al promedio global.	Garantizar una escala uniforme en los datos.
diferente_estacion	Indicador binario: 0 = misma estación de inicio y fin, 1 = estaciones distintas.	Identificar patrones de viajes circulares o unidireccionales.
viajes_por_dia	Número total de viajes realizados por día de la semana.	Analizar la **demanda diaria** del sistema.
bike_usage	Frecuencia de uso de cada bicicleta a lo largo del periodo de análisis.	Identificar bicicletas con **uso intensivo**.
duracion_tipo_prom	Duración promedio de los viajes según el tipo de pase (passholder_type).	Evaluar diferencias de comportamiento entre usuarios.

Las nuevas variables generadas enriquecen el dataset al capturar información crítica sobre **patrones temporales**, **eficiencia geoespacial** y **desempeño operativo**.

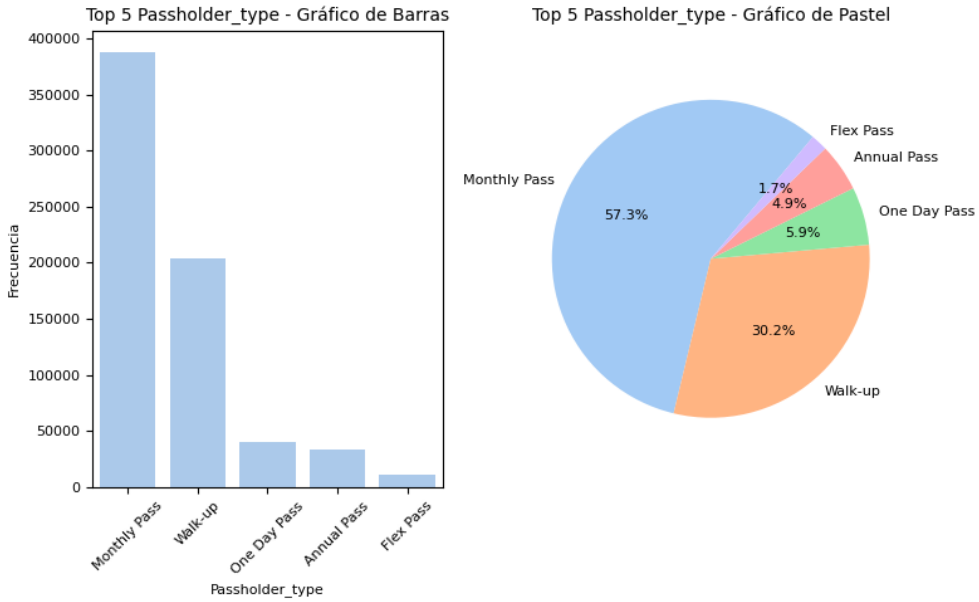
Categoría	**Variables Clave**
Temporales	franja_horaria, estacion_año, dia_semana, hora_inicio
Geoespaciales	distancia_mts
Desempeño/Agregadas	tiempo_promedio_estacion, tiempo_promedio_pares, velocidad_promedio_mts, duration_normalized, diferente_estacion, viajes_por_dia, bike_usage, duracion_tipo_prom

3. ANÁLISIS EXPLORATORIO (EDA)

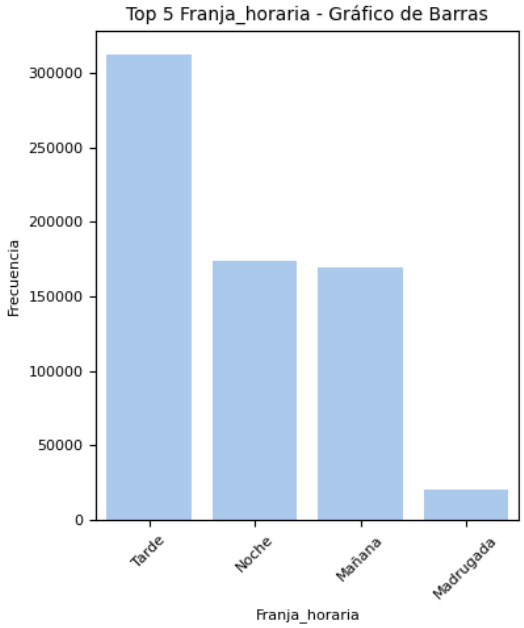
Patrones de Uso y Oportunidades de Optimización del Sistema de Bicicletas Compartidas

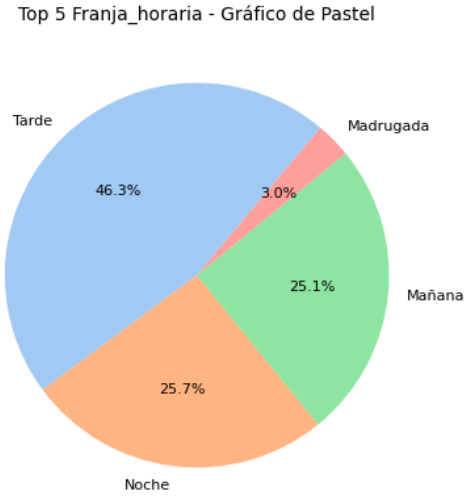


- Uso predominante del sistema: El 82.9% de los viajes son de "One Way", reflejando que los usuarios utilizan el sistema como complemento a otros medios de transporte. Solo un 17.1% corresponde a "Round Trip", lo que indica poca preferencia por recorridos circulares. Esto representa una oportunidad para crear incentivos que aumenten la adopción de estos viajes, especialmente en horarios y estaciones menos saturadas.

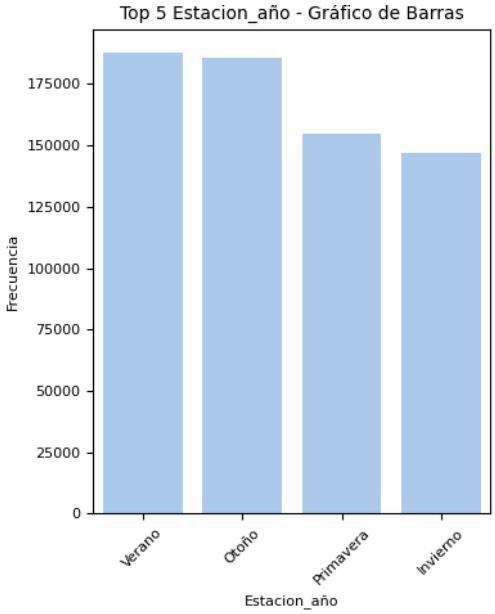


- Preferencia por tipos de pase: El "Monthly Pass" domina con un 57.3%, confirmando que la mayoría de los usuarios son recurrentes. El "Walk-up" (30.2%) capta a usuarios ocasionales, mientras que pases como "Annual Pass" (4.9%) y "Flex Pass" (1.7%) tienen poca adopción. Estrategias de fidelización y promociones específicas podrían atraer más usuarios a estos tipos de pase.





- Patrones de uso por franja horaria: La Tarde concentra el 46.3% de los viajes, asociada con el retorno a casa después del trabajo o estudio. Las franjas de Mañana (25.1%) y Noche (25.7%) también son relevantes, evidenciando un uso continuo a lo largo del día. La Madrugada, con apenas un 3.0%, representa una franja con baja demanda, ideal para optimizar la distribución del servicio.



- Estacionalidad del uso: Las estaciones de Verano (27.8%) y Otoño (27.5%) muestran la mayor demanda debido a condiciones climáticas favorables. En Invierno (21.8%), se observa una ligera caída, lo que sugiere una oportunidad para implementar promociones que mitiguen la estacionalidad y mantengan la demanda estable.

El alto uso del sistema en horarios y estaciones clave indica su papel como solución de última milla. Sin embargo, franjas como la Madrugada y pases poco utilizados son áreas de oportunidad. Optimizar la oferta en horarios críticos y fomentar el uso de otros tipos de pase permitirá un crecimiento más equilibrado y eficiente del servicio.

3.1 PRUEBA DE CHI-CUADRADO

Para evaluar la relación entre las variables categóricas y la variable objetivo **passholder_type**, se aplicó la **prueba estadística de Chi-Cuadrado**. Esta metodología permite identificar asociaciones significativas entre las características categóricas y la variable objetivo, proporcionando una base sólida para su inclusión en el modelo predictivo.

Variable	**Chi2**	**p-value**	**Interpretación**
trip_route_category	58,363.29	0.0000	Existe una fuerte asociación entre el tipo de ruta (One Way, Round Trip) y el tipo de pase.
franja_horaria	8,483.10	0.0000	La franja horaria muestra patrones claros en **horas pico** laborales y momentos de ocio.
estacion_año	2,726.49	0.0000	La estacionalidad influye en la demanda del sistema, con picos en verano y periodos vacacionales.

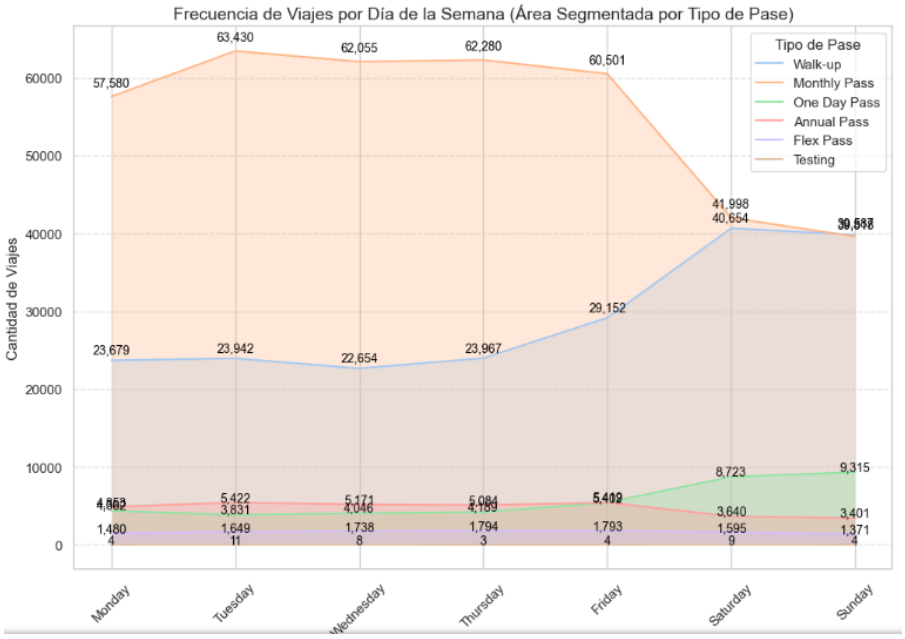
El análisis reveló que todas las variables categóricas evaluadas tienen una **relación estadísticamente significativa** con la variable objetivo **passholder_type** (p-value < 0.05). Su inclusión en el modelo está completamente justificada debido a la información crítica que aportan:

- **Preferencias de Uso**: La variable **trip_route_category** captura los patrones de preferencia de ruta según el tipo de pase.
- **Patrones Temporales**: Variables como **franja_horaria** y **estacion_año** permiten capturar la influencia de horarios pico y efectos estacionales en la demanda del sistema.

4. ANÁLISIS EXPLORATORIO DE VARIABLES NUMERICAS

Análisis de Frecuencia de Viajes: Día, Semana y Mes: El análisis de la frecuencia de viajes a lo largo del tiempo revela patrones importantes relacionados con **días de la semana**, **semanas del mes** y **meses del año**. Estos resultados permiten entender el comportamiento de los usuarios y cómo varía la demanda según el tipo de pase utilizado.

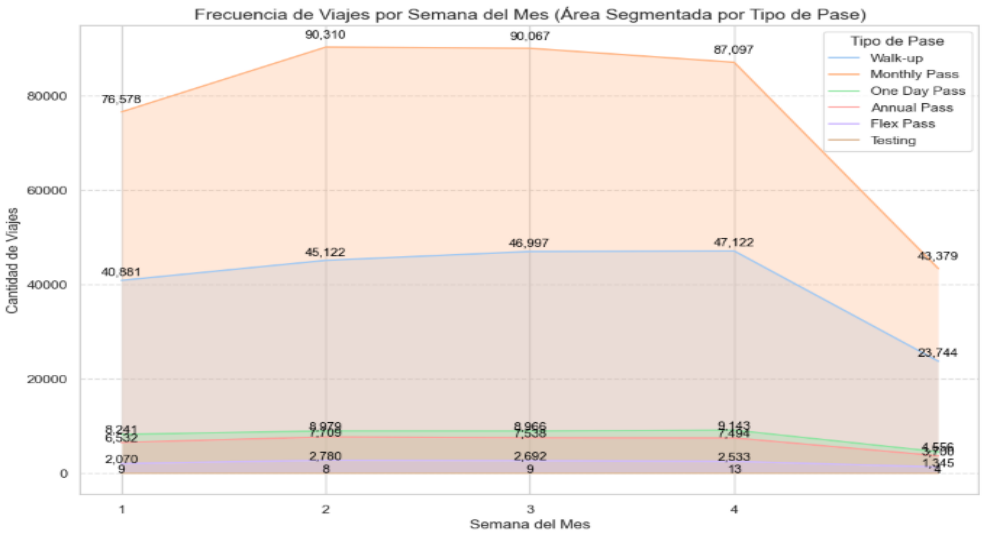
4.1Frecuencia de Viajes por Día de la Semana



Interpretación:

- El uso del sistema es más frecuente entre martes y jueves, con picos que superan los 63,000 viajes para los usuarios de "Monthly Pass".
- Lunes y viernes muestran una ligera caída, lo que podría estar asociado a días de menor actividad laboral.
- Los fines de semana destacan por un incremento en el uso de pases ocasionales como "Walk-up" y "One Day Pass", con un promedio cercano a 9,300 viajes los domingos.

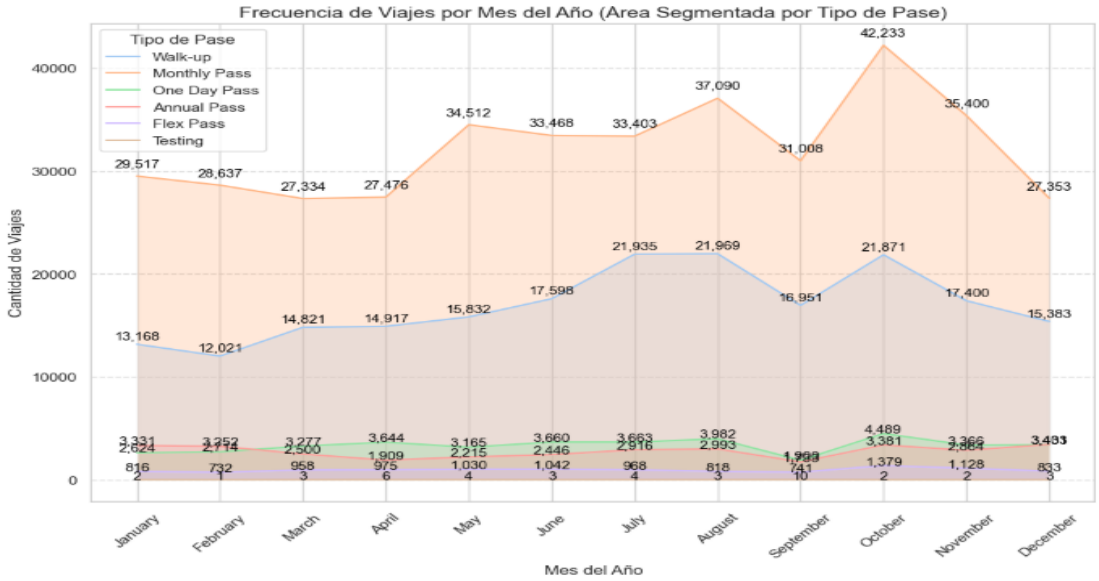
4.2 Frecuencia de Viajes por Semana del Mes



Interpretación:

- La segunda y tercera semanas del mes registran la mayor actividad, alcanzando picos de más de 90,000 viajes con predominancia de "Monthly Pass".
- Durante la cuarta semana, se observa una caída significativa en la frecuencia, disminuyendo a 43,379 viajes. Esta tendencia se replica también en los usuarios de "Walk-up", lo que indica un menor uso hacia finales del mes.

4.3 Frecuencia de Viajes por Mes del Año



Interpretación:

- Los meses de mayo, agosto y octubre presentan la mayor demanda, superando los 42,000 viajes mensuales en la categoría "Monthly Pass".
- Los meses de enero y febrero muestran los valores más bajos, con una reducción a 29,517 y 28,637 viajes, respectivamente.
- Los usuarios de "Walk-up" incrementan su actividad durante julio y diciembre, lo cual se alinea con periodos vacacionales y una mayor movilidad ocasional.
- "Monthly Pass": Dominante en todos los análisis (día, semana, mes), reflejando un uso constante y recurrente.
- "Walk-up" y "One Day Pass": Mayor actividad durante fines de semana, vacaciones y al final del mes.
- "Annual Pass" y "Flex Pass": Demanda muy baja en comparación con otros tipos de pases.

El sistema presenta patrones claros de alta demanda entre semana, especialmente en la segunda y tercera semanas del mes, y meses de mayo, agosto y octubre.
El uso por tipo de pase refleja que "Monthly Pass" es clave para la operación del sistema, mientras que los pases ocasionales como "Walk-up" y "One Day Pass" ganan relevancia en fines de semana y periodos vacacionales.

Para optimizar la demanda, se recomienda focalizar estrategias en los periodos de menor uso, como la cuarta semana del mes y los meses de enero y febrero.

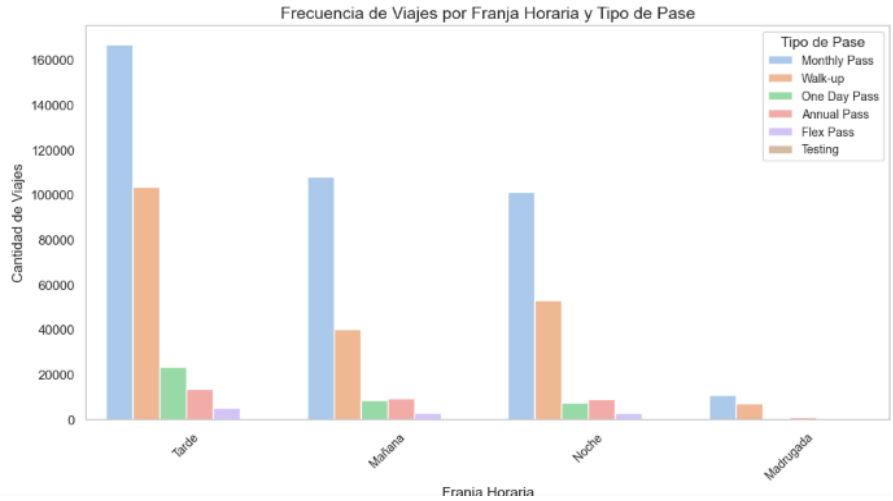
4.4 Análisis de Uso y Eficiencia de Estaciones

El análisis identifica las estaciones con mayor y menor uso del sistema:

- Estación 3005: La más utilizada con más de 27,000 viajes, destacándose como un punto clave de alta demanda.
- Estaciones 3014 y 3030: Ocupan el segundo y tercer lugar, con frecuencias entre 19,000 y 20,000 viajes, lo que sugiere su relevancia estratégica.
- Estaciones 4210 y 3082: Muestran una participación más baja dentro del Top 10, con 13,000 viajes, lo que puede estar relacionado con una menor conectividad o demanda local.

Este análisis evidencia una alta concentración de demanda en estaciones específicas, relacionadas posiblemente con zonas de mayor actividad económica o conexiones urbanas clave.

4.5 Frecuencia de Viajes por Franja Horaria y Tipo de Pase



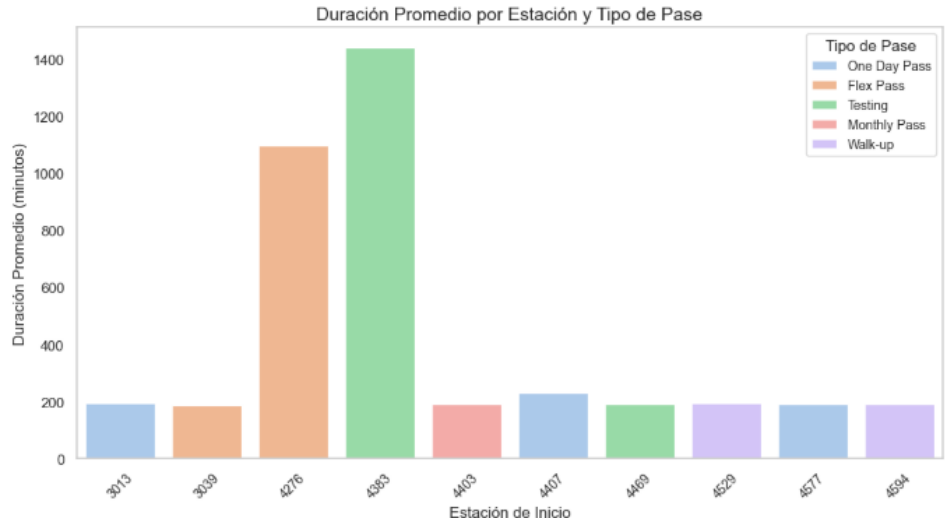
- Franja Tarde: Es la más concurrida con 160,000 viajes, impulsada principalmente por los usuarios de "Monthly Pass", reflejando la demanda en horarios laborales.

- Franja Mañana y Noche: Registra valores cercanos a 100,000 viajes, manteniendo la predominancia de "Monthly Pass".
- Franja Madrugada: La actividad es mínima, con apenas 10,000 viajes, concentrada en usuarios esporádicos como "One Day Pass".

Conclusión:
El uso del sistema varía significativamente a lo largo del día:

- "Monthly Pass" domina en todas las franjas, particularmente en horarios laborales.
- "Walk-up" y "One Day Pass" muestran mayor actividad en horarios no laborales, posiblemente vinculados a usos recreativos.

4.6 Duración Promedio por Estación y Tipo de Pase



- Duraciones Extremas:
    - La estación 4383 registra la mayor duración promedio con 1,400 minutos, seguida de 4276 con 1,100 minutos, asociadas a usos atípicos o anomalías del sistema.
- Duraciones Estándar:
    - Estaciones como 3013, 4403 y 4407 presentan duraciones promedio entre 200 y 250 minutos, reflejando un uso más eficiente y típico.
- Distribución Eficiente: Las estaciones con menores duraciones promedio indican un flujo constante de viajes.

Resumen General y Recomendaciones

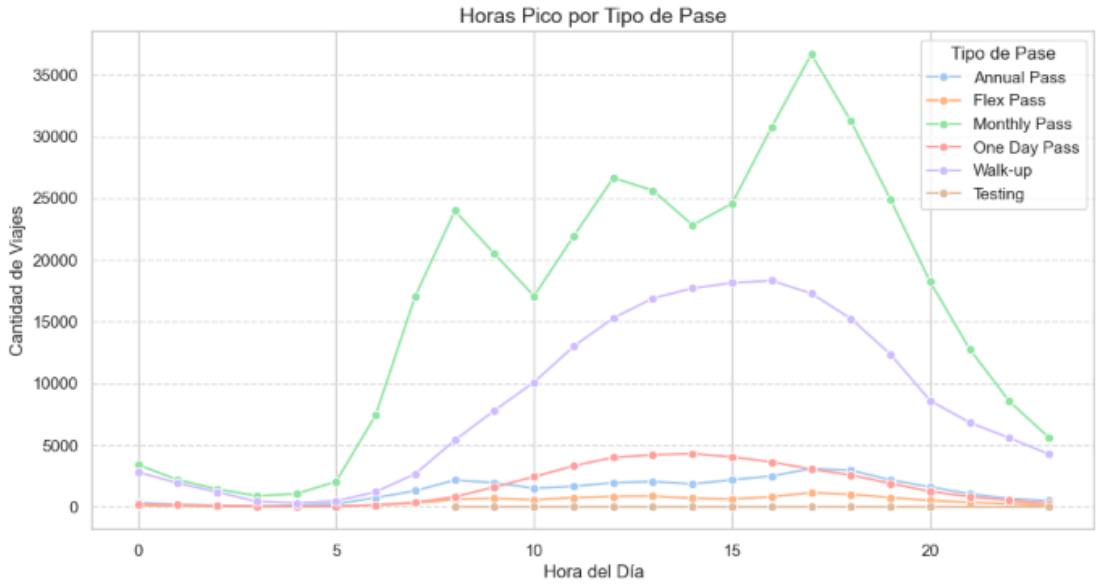
- La estación 3005 domina el sistema y representa un punto crítico de alta demanda.
- La franja Tarde es el periodo de mayor actividad, impulsada por los usuarios con "Monthly Pass".

- Se identifican anomalías operativas en estaciones con duraciones extremadamente altas, como 4383 y 4276, que requieren un análisis adicional.

Recomendaciones:

- Redistribuir bicicletas en estaciones de alta demanda para optimizar la oferta.
- Optimizar servicios durante franjas horarias menos utilizadas, como la madrugada.
- Evaluar estaciones con duraciones atípicas para detectar y corregir problemas operativos.

4.7 Análisis de Horas Pico y Comportamiento por Tipo de Pase



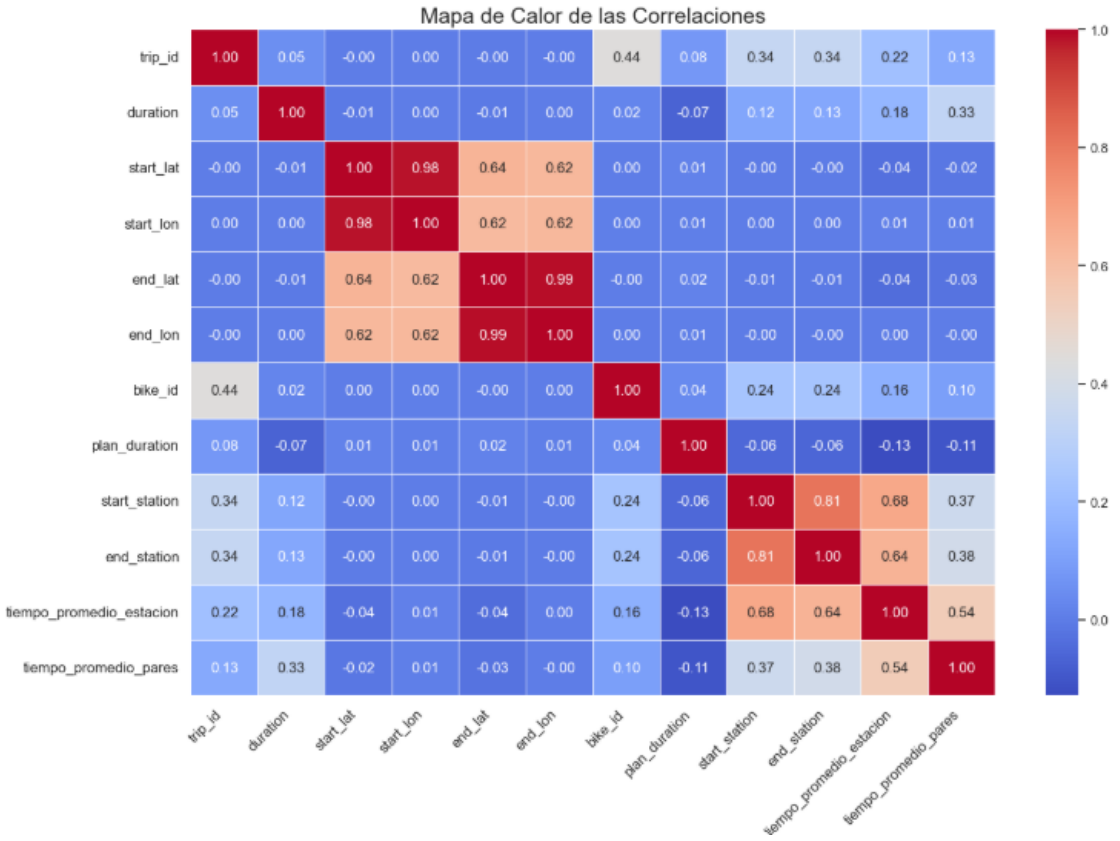
El análisis revela un patrón bimodal de demanda, con picos de viajes claramente definidos durante las horas laborales:

- Morning Peak (7:00 a 9:00 AM): Más de 24,000 viajes, asociados principalmente a desplazamientos laborales y estudiantiles.
- Evening Peak (4:00 a 6:00 PM): El pico más alto, con 36,000 viajes, reflejando el retorno a destinos habituales.

El "Monthly Pass" domina estos horarios pico, destacando su uso recurrente por parte de usuarios regulares. Por otro lado, los pases "Walk-up" y "One Day Pass" muestran una mayor actividad en horarios no pico, vinculada a actividades recreativas o esporádicas.

Recomendaciones: Se recomienda optimizar la capacidad operativa del sistema durante las horas pico (mañana y tarde) para atender la alta demanda. Además, se deben explorar estrategias para incentivar el uso en horarios no pico, donde la demanda es considerablemente menor, aprovechando el potencial de los usuarios ocasionales.

4.8 Análisis de Correlaciones y Selección de Variables



*El análisis del **mapa de correlaciones** permitió identificar relaciones significativas entre las variables del dataset.*

Correlaciones Fuertes

- start_lat y start_lon (0.98) y end_lat y end_lon (0.99): Alta correlación esperada debido a la relación geográfica de las estaciones de inicio y fin.
- start_station y end_station (0.81): Indica que los viajes suelen finalizar en la misma estación o en estaciones cercanas.
- tiempo_promedio_estacion y start_station (0.68): Relación importante que sugiere un patrón temporal en las estaciones de inicio y fin.

Correlaciones Moderadas

- duration y tiempo_promedio_pares (0.33): La duración del viaje tiene una relación moderada con los tiempos promedio entre pares de estaciones.
- bike_id y trip_id (0.44): Indica posibles patrones en la asignación de bicicletas a los viajes.

Correlaciones Débiles o Insignificantes

- Variables como **plan_duration** y **bike_id** muestran correlaciones cercanas a cero, por lo que su influencia en el modelo es mínima.

Columnas Relevantes para el Modelaje

A partir de este análisis, las siguientes variables se seleccionan por su importancia en la captura de patrones espaciales y temporales:

1. **start_station y end_station**: Capturan la relación entre estaciones de inicio y fin.
2. **duration**: Refleja patrones temporales de duración del viaje.
3. **start_lat, start_lon, end_lat, end_lon**: Información clave para capturar ubicaciones geográficas.
4. **tiempo_promedio_estacion y tiempo_promedio_pares**: Variables agregadas que aportan al análisis del desempeño en las estaciones y entre pares.

Próximo Paso: Análisis de Multicolinealidad

Se evaluará la **multicolinealidad** utilizando el **Factor de Inflación de Varianza (VIF)**. Este paso es crucial para identificar y ajustar variables redundantes con correlaciones extremadamente altas, garantizando así la estabilidad y robustez del modelo predictivo.

5. SELECCIÓN DE VARIABLES PARA EL MODELADO

En un inicio, se consideraron varias columnas del conjunto de datos para abordar tanto el análisis exploratorio como la construcción del modelo analítico. Estas variables fueron seleccionadas por su relevancia potencial para capturar patrones temporales, espaciales y de uso del servicio. Sin embargo, se realizó un análisis adicional para refinar esta selección con el objetivo de garantizar un modelado más estable y eficiente.

- **Análisis de Factor de Inflación de Varianza (VIF)**

Para asegurar la estabilidad y precisión del modelo, se llevó a cabo un análisis de multicolinealidad utilizando el Factor de Inflación de Varianza (VIF). Este análisis permitió identificar variables con alta redundancia entre sí, las cuales podrían comprometer la robustez del modelo.

Resultados del Análisis de VIF

Variable	VIF	Recomendación
trip_id	1.365268	Keep
duration	1.123660	Keep
start_lat	61.097853	Remove
start_lon	59.502005	Remove
end_lat	96.565162	Remove
end_lon	94.355805	Remove
bike_id	1.256399	Keep

plan_duration	1.032972	Keep
start_station	3.480485	Keep
end_station	3.195770	Keep
tiempo_promedio_estacion	2.686859	Keep
tiempo_promedio_pares	1.553129	Keep

- **Variables Excluidas**

Durante el análisis, se identificaron columnas que, aunque inicialmente seleccionadas, fueron descartadas por las siguientes razones:

A. Variables geoespaciales (start_lat, start_lon, end_lat, end_lon)
   - Presentaron valores de VIF superiores a 59, indicando una alta multicolinealidad.
   - La redundancia en la información geoespacial podría afectar la estabilidad del modelo.

B. plan_duration
   - Aunque su VIF era bajo, no estaba disponible en el conjunto de prueba (test), lo que imposibilitaba su uso para la predicción**.**

- **Variables Seleccionadas**

Se seleccionaron las siguientes variables por su bajo VIF, disponibilidad en ambos conjuntos (train y test) y relevancia estadística para la predicción de la variable objetivo **passholder_type**:

- trip_id: Identificador único del viaje.
- duration: Duración del viaje en minutos, clave para entender los patrones temporales.
- bike_id: Identificador de la bicicleta, útil para analizar la frecuencia de uso.
- start_station y end_station: Estaciones de inicio y fin, fundamentales para capturar los patrones espaciales.
- tiempo_promedio_estacion: Duración promedio de viajes iniciados en una estación.
- tiempo_promedio_pares: Duración promedio entre pares de estaciones.

Estas variables permiten capturar de manera eficiente:

1. Patrones temporales y de duración: A través de duration y variables agregadas como tiempo_promedio_estacion.
2. Patrones espaciales: Mediante el uso de start_station y end_station.

3. Frecuencia y uso del sistema: Mediante identificadores únicos como trip_id y bike_id.

La selección de variables se realizó de manera **a priori**, eliminando aquellas con alta multicolinealidad y poca disponibilidad. Esta primera versión del conjunto de características será utilizada como base para el proceso de modelado.

Sin embargo, en etapas posteriores, se evaluarán **posibles transformaciones** adicionales de las variables seleccionadas, tales como:

- Creación de variables derivadas para capturar nuevas relaciones.
- Análisis de interacción entre variables.
- Reducción de dimensionalidad, en caso de ser necesario.

Este enfoque garantiza un balance entre simplicidad y representatividad del modelo, facilitando un proceso iterativo que permitirá optimizar su desempeño y capacidad predictiva.

Variables Seleccionadas para el Modelado

Tras refinar la selección inicial, se identificaron las siguientes columnas como las más relevantes para abordar el problema planteado. Estas variables cumplen con criterios de bajo VIF, disponibilidad en ambos conjuntos (train y test), y relevancia estadística:

- trip_id: Identificador único del viaje. Útil para la trazabilidad de registros.
- duration: Duración del viaje en minutos, clave para entender los patrones temporales.
- bike_id: Identificador de la bicicleta, relevante para analizar la frecuencia de uso.
- start_station y end_station: Estaciones de inicio y fin, fundamentales para capturar patrones espaciales.
- tiempo_promedio_estacion: Duración promedio de viajes iniciados en cada estación, indicando su popularidad y tiempo de uso.
- tiempo_promedio_pares: Duración promedio entre pares de estaciones, proporcionando información clave sobre trayectorias comunes.

6. PROCESO DE ENTRENAMIENTO Y EVALUACIÓN DEL MODELO

El proceso inició con un **preprocesamiento estructurado** utilizando un pipeline para garantizar la **automatización y consistencia** en el tratamiento de los datos. Los pasos clave incluyeron:

- **Transformación de características**:
  Se utilizaron transformaciones específicas para variables numéricas y categóricas mediante un **ColumnTransformer**.

- **Reducción de dimensionalidad con PCA**: Implementamos **PCA (Análisis de Componentes Principales)** para reducir la cantidad de atributos, optimizar la eficiencia de los modelos y evitar problemas de multicolinealidad.

Este enfoque permitió obtener un conjunto de características optimizado y listo para el proceso de modelado.

6.1 Modelos Evaluados

Se seleccionaron y evaluaron cuatro modelos iniciales para comparar su rendimiento en la predicción de la variable passholder_type:

- **Regresión Logística**
- **Árbol de Decisión**
- **Random Forest**
- **K-Nearest Neighbors (KNN)**

6.2 Métricas de Evaluación

Para garantizar una evaluación justa y completa de los modelos, utilizamos las siguientes métricas:

1. **Accuracy**
    - Mide el **porcentaje de predicciones correctas** sobre el total de observaciones.
    - Es útil en problemas con clases balanceadas, pero puede ser insuficiente si existe un desbalance de clases.
2. **Precision (Weighted)**
    - Calcula la proporción de predicciones positivas que realmente son correctas, ponderada por clase.
    - Es clave para **minimizar falsos positivos**.
3. **Recall (Weighted)**
    - Indica la capacidad del modelo para identificar correctamente las instancias positivas, ponderada por clase.
    - Es crucial en problemas donde **los falsos negativos son costosos**, como en nuestro caso operativo.
4. **F1-Score (Weighted)**
    - Es la **media armónica entre Precision y Recall**, ponderada por clase.
    - Proporciona un equilibrio entre ambas métricas.

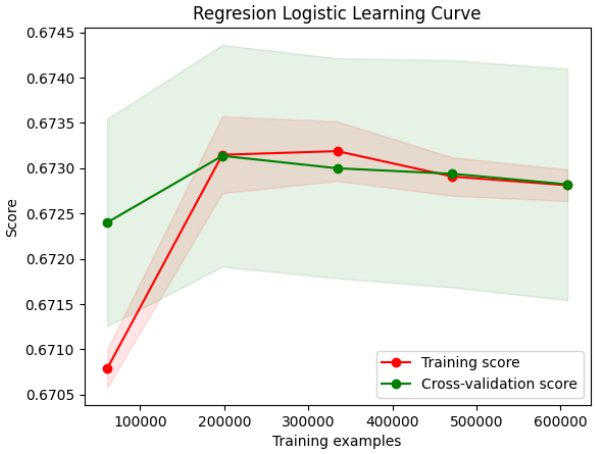
6.3 Resultados Iniciales

Los puntajes obtenidos por cada modelo en las métricas mencionadas fueron los siguientes:

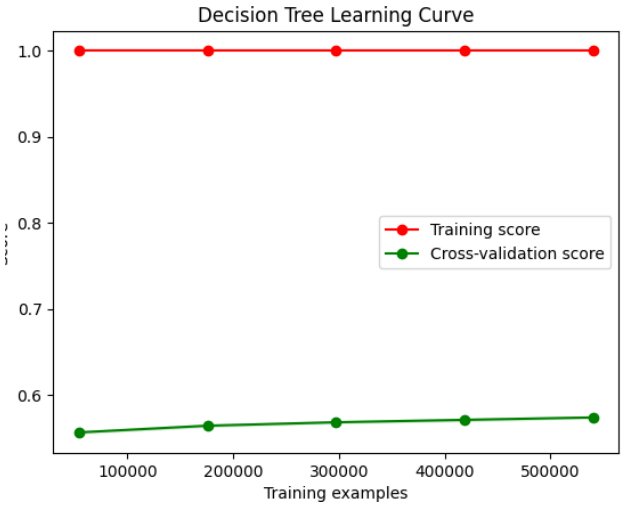
Modelo	Accuracy	Precision (weighted)	Recall (weighted)	F1-Score (weighted)
Logistic Regression	67.1%	62.4%	**67.1%**	61.3%
Decision Tree	57.4%	57.9%	57.4%	57.7%
Random Forest	**70.2%**	68.2%	**70.2%**	65.6%
K-Nearest Neighbors	65.9%	62.5%	65.9%	63.2%

6.4 Análisis de las Curvas de Aprendizaje

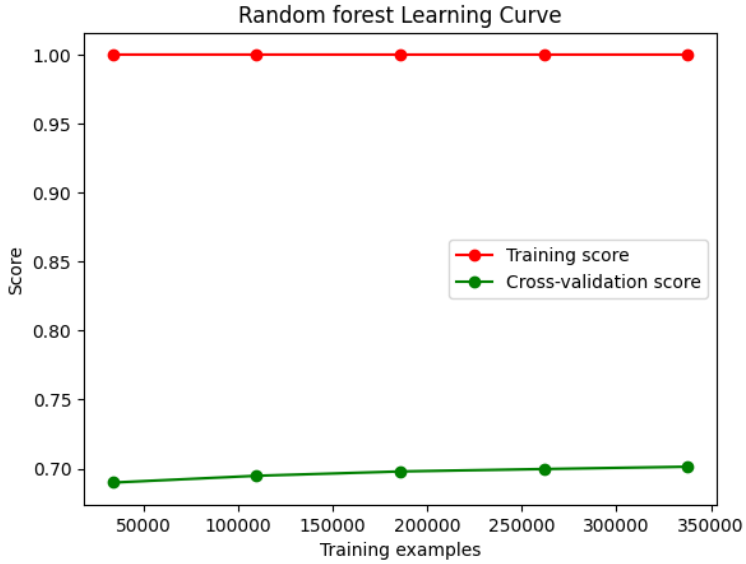
Se analizaron las curvas de aprendizaje de cada modelo para identificar posibles problemas de sobreajuste o bajo rendimiento:



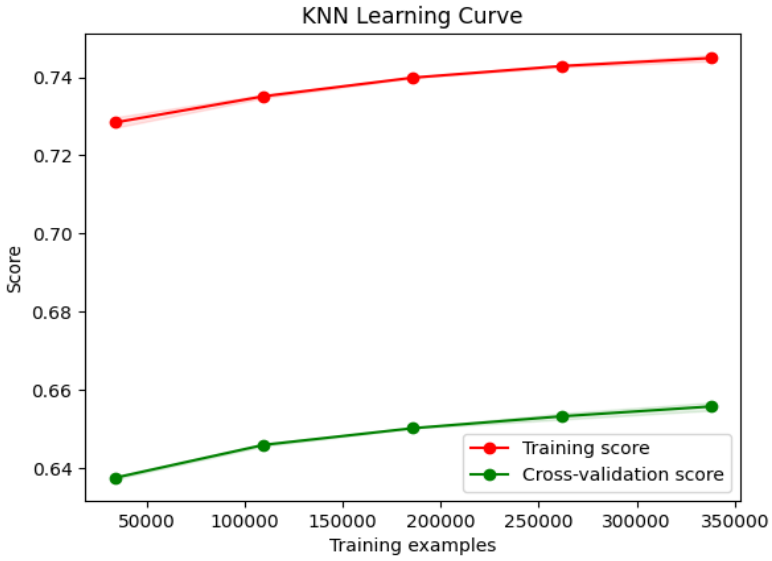
- Regresión Logística:
    - Mostró un equilibrio entre la training score y cross-validation score, lo que indica buen ajuste y ausencia de sobreentrenamiento.



- Árbol de Decisión:
    - Exhibió un claro sobreajuste: puntuación perfecta en el conjunto de entrenamiento, pero bajo rendimiento en validación.



- Random Forest:
    - Aunque su desempeño fue competitivo, presentó una brecha significativa entre las curvas, lo que sugiere un leve sobreajuste.



- KNN:
    - Mejoró gradualmente con más datos, pero no logró cerrar la brecha entre las curvas.

6.5 Selección del Modelo: Regresión Logística

La Regresión Logística fue seleccionada a priori como modelo final debido a los siguientes factores:

- Estabilidad en las curvas de aprendizaje, sin signos de sobreajuste.
- Recall ponderado del 67.1%, lo que garantiza la correcta identificación de la mayoría de las instancias positivas.
- Equilibrio entre las métricas, mostrando un rendimiento sólido y consistente.

6.6 Optimización del Modelo

Se realizó un tuning de hiperparámetros con GridSearchCV para mejorar aún más el desempeño del modelo, con un enfoque en Recall. Esto es fundamental porque minimizar falsos negativos es prioritario en la planificación operativa del servicio.

- Mejores parámetros obtenidos:
    - C (Regularización): 1.0
    - Penalty: l2
    - Solver: newton-cg

La optimización permitió mantener el Recall alto sin comprometer el equilibrio general del modelo.

6.7 Conclusión del Proceso

Se implementó un pipeline completo para el **preprocesamiento y modelado** de datos.

- Se compararon múltiples modelos utilizando **métricas clave** y análisis de **curvas de aprendizaje**.
- La **Regresión Logística** fue seleccionada como el modelo final debido a su:
- **Estabilidad** y equilibrio en las métricas.
- Capacidad para priorizar el **Recall** y reducir falsos negativos.
- El **tuning de hiperparámetros** optimizó aún más el modelo, consolidando su desempeño para implementación en producción.

Este proceso asegura un modelo confiable, robusto y alineado con los objetivos operativos del servicio de bicicletas compartidas.

7. PROCESO DE ENTRENAMIENTO: XGBOOST COMO MODELO FINAL

Luego de evaluar múltiples modelos como **Regresión Logística, Árboles de Decisión, Random Forest y KNN**, se decidió probar un enfoque más avanzado con **XGBoost**. Este modelo fue seleccionado por su capacidad de manejar grandes volúmenes de datos, **eficiencia computacional** (con soporte para GPU) y su habilidad para encontrar patrones complejos en datos desbalanceados.

7.1 Configuración Inicial del Modelo

El modelo **XGBoost** se configuró con los siguientes parámetros iniciales:

- **tree_method**: 'hist' (para aprovechar la aceleración en GPU).
- **device**: 'cuda:0' (ejecución eficiente en GPU).
- **objective**: 'multi:softprob' (problema de clasificación multiclase).
- **eval_metric**: 'mlogloss' (log-loss como métrica de evaluación).

7.3 Entrenamiento y Evaluación Inicial

Tras entrenar el modelo con los datos de entrenamiento, se obtuvo un **accuracy inicial del 72.08%**, demostrando un mejor desempeño en comparación con los modelos previos. Sin embargo, se identificaron áreas de mejora en términos de **Recall** para ciertas clases, particularmente aquellas con menor representación.

7.3 Optimización del Modelo (Tuning)

Se implementó RandomizedSearchCV para realizar un tuning de hiperparámetros, buscando mejorar el rendimiento del modelo sin incurrir en sobreajuste. Los parámetros evaluados incluyeron:

- Número de árboles (n_estimators): [100, 200, 300]
- Profundidad máxima del árbol (max_depth): [3, 5, 7]
- Tasa de aprendizaje (learning_rate): [0.01, 0.1, 0.2]
- Submuestreo (subsample): [0.8, 1.0]
- Proporción de características por árbol (colsample_bytree): [0.8, 1.0]

Mejores parámetros encontrados:

Parámetro	Valor
n_estimators	300
max_depth	7
learning_rate	0.2
subsample	0.8
colsample_bytree	0.8
tree_method	hist
device	cuda:0

7.4 Resultados Finales

Métrica	Valor
Accuracy	73%
Macro Recall	34%
Macro Precision	77%
Wighted F1 Score	70%

7.5 Reporte de clasificación:

- Clases más fáciles de predecir:
  Clases con mayor representación, como "Monthly Pass" y "Walk-up", alcanzaron altos valores de Recall y Precision.
- Clases difíciles de predecir:
  Las clases con menor representación, como "Annual Pass" y "Flex Pass", presentaron bajo desempeño en Recall debido al desbalance de clases.

7.6 Justificación de la Selección del XGBoost

El modelo XGBoost fue seleccionado como el ganador por las siguientes razones:

A. Mejor Rendimiento General:
   - Logró el mejor equilibrio entre las métricas, con un Accuracy del 73% y un Weighted F1-Score del 70%.
   - Su capacidad para manejar características complejas y patrones en datos desbalanceados superó a los modelos anteriores.

B. Robustez y Generalización:
   - El uso de técnicas de regularización y parámetros optimizados permitió evitar el sobreajuste, manteniendo un rendimiento estable en validación.

C. Manejo de Datos Desbalanceados:
   - Aunque se consideraron técnicas de balanceo de clases (como SMOTE o clase ponderada), no se implementaron por precaución de sobrefitting.
   - El modelo logró adaptarse a la naturaleza desbalanceada de los datos con RandomizedSearchCV y tuning adecuado.

7.7 Observaciones Finales

- Clases Desbalanceadas:
  El desbalance de clases dificultó la predicción de algunas categorías menos representadas.
  El modelo podría mejorarse aún más mediante técnicas de reponderación o aumento de datos sintéticos.

- Modelo Generalizable:
  XGBoost, con su eficiencia computacional y optimización, se adaptó mejor a los datos complejos, resultando en un modelo robusto y generalizable.

8. CONCLUSIÓN

El modelo **XGBoost** fue seleccionado como el mejor clasificador, logrando un equilibrio sólido entre Recall y Precision en un escenario con clases desbalanceadas. Su capacidad para manejar grandes volúmenes de datos, encontrar patrones complejos y evitar el sobreajuste lo convierte en la solución ideal para la predicción del tipo de pase (passholder_type). Este modelo ahora se encuentra listo para su implementación en producción.

9. PROCESO DE DESPLIEGUE

Para desplegar PEDAL INSIGHTS AI en Amazon Web Services, utilizaremos los siguientes servicios principales:

Servicios para utilizar

- Amazon Elastic Compute Cloud (EC2)
  - Crearemos una instancia EC2 con Ubuntu, donde desplegaremos el contenedor Docker que contiene nuestra aplicación.
  - Configuraremos los **Security Groups** para abrir el **puerto 7860**, lo que permitirá el acceso externo a la aplicación.
- Amazon Elastic Container Registry (ECR)
  - Usaremos ECR como nuestro registro privado para almacenar la imagen Docker de **PEDAL INSIGHTS AI**.

Pasos para el despliegue: *Subir la imagen a Amazon ECR*, Primero, construiremos la imagen Docker en nuestro entorno de desarrollo.
En la terminal, ejecutaremos el siguiente comando:

- *Crear un repositorio en ECR*:

```
aws ecr create-repository --repository-name pedal-insights-ai
```

- *Autenticarnos en ECR: Ejecutaremos el siguiente comando para obtener las credenciales necesarias*

Script: aws ecr get-login-password --region <region> | docker login --username AWS --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com

- *Etiquetar la imagen Docker*

Script:docker tag pedal-insights-ai<aws_account_id>.dkr.ecr.<region>.amazonaws.com/pedal-insights-ai

- Subir la imagen a ECR

Script: docker push <aws_account_id>.dkr.ecr.<region>.amazonaws.com/pedal-insights-ai

- *Crear la instancia EC2 (Ubuntu)*

Para desplegar la aplicación, crearemos una instancia EC2 con **Ubuntu 22.04**. Durante la configuración:

  - Seleccionaremos un tamaño adecuado, por ejemplo, una instancia t2.medium.
  - Configuraremos un **Security Group** para abrir el **puerto 7860**, permitiendo conexiones desde cualquier IP (0.0.0.0/0).

- *Instalar Docker en la instancia EC2: Nos conectaremos a la instancia EC2 mediante SSH y ejecutaremos los siguientes comandos para instalar Docker*

```
sudo apt update
sudo apt install -y docker.io
sudo systemctl start docker
sudo systemctl enable docker
```

- *Ejecutar el contenedor desde ECR*

Script: sudo docker run -p 7860:7860 <aws_account_id>.dkr.ecr.<region>.amazonaws.com/pedal-insights-ai

- ***Configurar el Security Group***

*Para permitir el acceso externo a la aplicación, configuraremos el **Security Group**:*

  - *Abriremos el **puerto 7860**.*
  - *Permitiremos el acceso desde cualquier IP (0.0.0.0/0).*

- Acceder a la aplicación: Finalmente, accederemos a **PEDAL INSIGHTS AI** utilizando la IP pública de la instancia EC2.
  La URL será -> http://<PUBLIC_IP>:7860